1.   A packet for transferring data in a load/store fabric
     to a shared input/output (I/O) endpoint, comprising:

     a header field; for identifying the shared I/O
          endpoint; and

     an OS Domain header field, included within said header
          field, for identifying which one of a plurality
          of root complexes is associated with the packet.

2.   The packet as recited in claim 1 wherein each of said
     plurality of root complexes comprises an operating
     system domain (OSD).

3.   The packet as recited in claim 2 wherein said
     operating system domain comprises:

     a processing complex; and

     a memory, coupled to said processing complex, for
          storing data utilized by said processing complex.

4.   The packet as recited in claim 1 wherein at least one
     of said plurality of root complexes comprises a
     plurality of operating system domains.

5.   The packet as recited in claim 4 wherein said
     plurality of operating system domains comprise:

     a plurality of processing complexes, each having a
          memory coupled to them for storing data utilized
          by them.

6.   The packet of claim 1 wherein the data comprises information which is desired to be transferred from one of said plurality of root complexes to the shared I/O endpoint.

7.   The packet of claim 1 wherein said information comprises command/message information.

8.   The packet of claim 1 wherein the data comprises information which is desired to be transferred from the shared I/O endpoint to one of said plurality of root complexes.

9.   The packet of claim 1 wherein the load/store fabric is hardware, software, or a combination of hardware and software that moves the data from one of said plurality of root complexes to the shared I/O endpoint.

10.   The packet of claim 1 wherein the shared input/output (I/O) endpoint comprises a network interface controller (NIC).

11.   The packet of claim 10 wherein said network interface controller is an Ethernet controller.

12.   The packet of claim 1 wherein the shared input/output (I/O) endpoint comprises a Fiber Channel controller.

13.   The packet of claim 1 wherein the shared input/output (I/O) endpoint comprises a shared RAID controller.

14.   The packet of claim 1 wherein the load/store fabric utilizes PCI Express.

15. The packet of claim 14 wherein said header field comprises:

a transaction layer packet (TLP) header; and

an optional data payload.

16. The packet of claim 15 wherein said TLP header comprises a plurality of fields, including a field identifying the shared I/O endpoint for which the packet is destined.

17. The packet of claim 1 wherein said OS Domain header field comprises an OS Domain Number field.

18. The packet of claim 17 wherein said OS Domain Number field is global throughout the load/store fabric.

19. The packet of claim 17 wherein said OS Domain Number field is local to each link in the load/store fabric.

20. The packet of claim 17 wherein said OS Domain Number field specifies one of said plurality of root complexes from which the packet originated.

21. The packet of claim 17 wherein said OS Domain Number field specifies one of a plurality of OS Domains from which the packet originated.

22. The packet of claim 17 wherein said OS Domain Number field is a six (6) bit field for designating up to 64 distinct root complexes.

23. The packet of claim 21 wherein said OS Domain Number field is a six (6) bit field for designating up to 64 distinct OS Domains within a link in the load/store fabric.

24. The packet of claim 1 wherein said load/store fabric is a serial load/store fabric.

25. The packet of claim 1 wherein said load/store fabric is a bus.

26. An OS Domain header, within a PCI Express Packet comprising:

a plurality of bits, said plurality of bits defining an operating system domain from which the PCI Express Packet originated.

27. The OS Domain header as recited in claim 26 wherein said operating system domain comprises a root complex.

28. The OS Domain header as recited in claim 26 wherein said operating system domain comprises:

a processing complex; and

a memory, coupled to said processing complex for storing data utilized by said processing complex.

29. The OS Domain header as recited in claim 26 wherein said operating system domain comprises a port within a shared I/O switch to which a root complex is coupled.

30. The OS Domain header as recited in claim 26 wherein the OS Domain header is attached to a PCI Express Packet to form a PCI Express+ Packet.

31. The OS Domain header as recited in claim 30 wherein said PCI Express+ Packet is sent to an endpoint device.

32. The OS Domain header as recited in claim 31 wherein said endpoint device is a shared I/O network interface controller.

33. The OS Domain header as recited in claim 32 wherein said shared I/O network interface controller, upon receipt of said PCI Express+ Packet, examines the OS Domain header to determine which operating system domain the PCI Express Packet originated from.

34. The OS Domain header as recited in claim 33 wherein said shared I/O network interface controller, after determining which operating system domain the PCI Express Packet originated from, applies controller resources that are associated with that operating system domain.

35. The OS Domain header as recited in claim 34 wherein said controller resources comprise:

a plurality of controller register sets; and

a plurality of direct memory access (DMA) engines.

36. A method for identifying a root complex for a packet
     within a load/store fabric to allow for sharing of
     input/output (I/O) endpoints, the method comprising:

     providing an architecture for the packet; and

     providing a field for inclusion in the packet to
          identify the root complex for the packet;

     wherein the input/output (I/O) endpoints utilize the
          field provided in said step of providing a field
          to identify the root complex for the packet.

37. The method for identifying the root complex as recited
     in claim 36 wherein the root complex comprises a
     network computer server.

38. The method for identifying the root complex as recited
     in claim 37 wherein the network computer server is a
     blade server.

39. The method for identifying the root complex as recited
     in claim 36 wherein the root complex comprises an
     operating system domain.

40. The method for identifying the root complex as recited
     in claim 36 wherein the architecture in said step of
     providing an architecture conforms to the PCI Express
     System Architecture.

41. The method for identifying the root complex as recited
     in claim 36 wherein the input/output (I/O) endpoints
     comprise a network interface controller.

42. The method for identifying the root complex as recited in claim 41 wherein the network interface controller is an Ethernet controller.

43. The method for identifying the root complex as recited in claim 36 wherein the input/output (I/O) endpoints comprise a disk storage controller.

44. The method for identifying the root complex as recited in claim 43 wherein the disk storage controller is a Fiber Channel controller.

45. The method for identifying the root complex as recited in claim 43 wherein the disk storage controller is a serial ATA controller.

46. A method for transferring a packet from a shared input/output (I/O) endpoint to one of a plurality of OS Domains, within a load/store fabric, comprising:

 embedding an OS Domain number with the packet to associate the packet with one of the plurality of OS Domains;

 transferring the packet with the embedded OS Domain number to a shared I/O switch;

 examining the embedded OS Domain number to determine a port within the shared I/O switch associated with the one of the plurality of OS Domains; and

 transferring the packet to the one of the plurality of OS Domains using the port.

47.   The method as recited in claim 46 wherein the shared input/output (I/O) endpoint comprises a network interface controller.

48.   The method as recited in claim 46 wherein each of the plurality of OS Domains comprise:

a processing complex; and

memory, coupled to the processing complex for storing data utilized by the processing complex.

49.   The method as recited in claim 48 wherein the processing complex comprises one or more processors.

50.   The method as recited in claim 46 wherein the load/store fabric utilizes PCI Express.

51.   The method as recited in claim 46 wherein the load/store fabric maps memory space for the shared I/O switch within memory space of the one of the plurality of OS Domains.

52.   The method as recited in claim 46 wherein said step of embedding comprises:

forming an OS Header field; and

including the OS Header field within the packet for transfer to the shared I/O switch.

53.   The method as recited in claim 46 wherein said step of examining comprises:

performing a table lookup to associate the OS Domain number with a PCI bus hierarchy for the packet; and

determining a port associated with the PCI bus hierarchy which is coupled to the OS Domain for the PCI bus hierarchy.

54. The method as recited in claim 46 wherein the port couples the shared I/O switch to the one of the plurality of OS Domains.